

IS administrace databázového výukového portálu

IS for Administration of Database Educational Portal

Zadání bakalářské práce

Student: **Michal Dirgas**

Studijní program: B2647 Informační a komunikační technologie

Studijní obor: 2612R025 Informatika a výpočetní technika

Téma: **IS administrace databázového výukového portálu**
IS for Administration of Database Educational Portal

Zásady pro vypracování:

Katedra informatiky poskytuje výuku pěti předmětů v oblasti databázových systémů. Na podporu výuky byl vytvořen webový portál dbedu.cs.vsb.cz, který poskytuje studentovi studijní materiály z přednášek i cvičení, instalační soubory, skripta, různá videa, manuály a podobně. Portál taktéž studenty informuje o novinkách v předmětech v průběhu semestru jako jsou například termíny zápočtů, zrušené přednášky a jiné. Informace na portálu se aktualizují každý týden, což je kvůli absenci webové administrace časově náročné. Cílem této práce je vytvoření webové administrace tohoto portálu.

1. Seznámit se se stávajícím webovým portálem.
2. Nastudovat přihlášení uživatele do systému pomocí LDAP a definování práv jednotlivých uživatelů.
3. Nastudovat problematiku správy souborového systému pomocí webové aplikace.
4. Nastudovat problematiku editace xml a html souborů pomocí webové aplikace.
5. Navrhnout, naimplementovat a otestovat IS pro administraci webového portálu.

Seznam doporučené odborné literatury:

Podle pokynů vedoucího bakalářské práce.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Peter Chovanec**

Datum zadání: 01.09.2013

Datum odevzdání: 07.05.2015



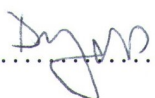
doc. Dr. Ing. Eduard Sojka
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne. Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

V Ostrave 7. mája 2015

.....


Týmito riadkami by som rád poďakoval svojmu vedúcemu bakalárskej práce Ing. Petrovi Chovancovi za pomoc pri riešení zložitejších problémov, a taktiež pri vedení a vytváraní tejto bakalárskej práce. Ďalej by som rád poďakoval Ing. Pavlovi Bednářovi za rady a pomoc pri implementácii a poskytnutí komponent počas vývoja práce. V poslednej rade ďakujem rodine a priateľom za podporu počas celého štúdia.

Abstrakt

Informačný systém je kombinácia technológií, informácií a ľudí, ktorý zabezpečuje ukladanie, spracovávanie a ďalšie šírenie týchto dát. Cieľom tejto práce je vylepšenie a rozšírenie informačného systému na správu výučby databázových predmetov. V informačnom systéme sú vytvorené štyri typy užívateľov: administrátor, učiteľ, študent a anonymný hosť. Každá užívateľská rola má vlastnú sekciu v rámci informačného systému a taktiež vlastné prístupové práva a možnosti používania. Aktualizovaný informačný systém prináša zefektívnenie štúdia pre študentov a výučby pre vyučujúcich.

Kľúčové slová: Informačný systém, ASP.NET, DB Education, ELFinder, CKEditor

Abstract

Information system is a combination of technologies, information and people, that ensures the storing, processing and other spreading of these data. The aim of this bachelor thesis is the improvement and extension of an information system for administration of education of database subjects. In the information system four types of users are created: administrator, teacher, student and guest. Each user role has its own section within the information system and also own access rights and possibilities for use. Updated information system is more effective for students to study and for tutors to teach.

Keywords: Information system, ASP.NET, DB Education, ELFinder, CKEditor

Zoznam použitých skratiek a symbolov

AJAX	– Asynchronous JavaScript and XML
ASP	– Active Server Pages
API	– Application Programming Interface
CSS	– Cascading Style Sheets
DB	– Databáza
DOM	– Domain Object Model
FTP	– File Transfer Protocol
GUI	– Graphical User Interface
HTML	– HyperText Markup Language
IS	– Informačný systém
JS	– JavaScript
LDAP	– Lightweight Directory Access Protocol
LINQ	– Language-Integrated Query
MAC	– Macintosh
PDF	– Portable Document Format
PHP	– Hypertext Preprocessor
RDC	– Remote Desktop Connection
RTF	– Rich Text Format
SMTP	– Simple Mail Transfer Protocol
SQL	– Structured Query Language
WebDAV	– Web Distributed Authoring and Versioning
WYSIWYG	– What You See Is What You Get
XHTML	– Extensible Hypertext Markup Language
XML	– Extensible Markup Language

Obsah

1	Úvod	4
2	Informačný systém DB Education	5
2.1	Pôvodný informačný systém	5
2.2	Ciele bakalárskej práce	5
3	Analýza	6
3.1	Datová analýza	6
3.2	Užívateľské role	8
3.3	Diagramy užívateľských rolí	9
3.4	Funkčná analýza	11
3.5	Detailný popis funkcií	13
4	Použité technológie	15
4.1	ASP.NET	15
4.2	SQLite	15
4.3	Pôvodný editor	16
4.4	HTML/XML editory	16
4.5	Správcovia súborov	22
4.6	Zhrnutie a výber komponent	26
5	Implementácia	27
5.1	Webové rozhranie	27
5.2	Použité komponenty	30
5.3	Úprava zdrojového kódu ELFinderu	32
5.4	Udalosti v Menu.ascx	33
5.5	Kontrola odosielaných súborov v .zip archíve	36
6	Záver	39
7	Literatúra	40
	Prílohy	41
A	Obsah priloženého CD	42

Zoznam obrázkov

1	ER-Diagram systému DB Education	7
2	Use-case diagram pre rolu Admin a Teacher	9
3	Use-case diagram pre rolu Študent a Guest	10
4	Vzhľad komponenty Ajax Control Toolkit	17
5	Vzhľad komponenty XML WebPad Editor	18
6	Vzhľad komponenty CKEditor	19
7	Vzhľad komponenty CuteEditor/RichEditor	20
8	Vzhľad komponenty RadEditor	21
9	Vzhľad komponenty HttpCommander	23
10	Vzhľad komponenty CKFinder	24
11	Vzhľad komponenty ELFinder	25
12	Informačný systém DB Education	27
13	Menu DB Education	28
14	Správa webovej stránky	29
15	Ukážka informačného emailu	31

Zoznam výpisov zdrojového kódu

1	Obmedzenie prístupu na webovú stránku	30
2	Získanie celého mena pomocou komponenty VsbLogin	30
3	Potvrdzovací email	31
4	Upravená metóda GetRootDirectory	32
5	Upravená metóda v triede OpenCommand	33
6	Spojenie s databázou	33
7	Volanie udalosti v Menu.ascx	34
8	Naviazaná metóda z masterpage	35
9	Časť metódy UnzipAndCheck	36
10	Porovnanie požadovaných súborov	37

1 Úvod

V dnešnej dobe je informačný systém štandard všade, kam sa pozrieme. Môžeme nájsť systémy, ktoré automaticky riadia beh a správne fungovanie prevádzky, či už u firiem, škôl alebo hocikde inde v priemysle. V tejto práci sa venujem rozšíreniu a vylepšeniu informačného systému DB Education pre výuku databázových predmetov. Pôvodný informačný systém bol neefektívny, či už v administrácii, alebo aj vo výučbe.

Na začiatku práce sa venujem podrobnej analýze celého informačného systému. Popisujem ako medzi sebou fungujú jednotlivé komponenty, kto a ako využíva IS, čo sú hlavné možnosti správy a používania informačného systému, a iné.

Neskôr približujem použité technológie, ako napr. ASP.NET WebForms, alebo SQLite databázu. Jednou z požiadaviek pre túto prácu bola integrácia prvkov pre správu a editáciu súborov. V ďalšej kapitole sa preto venujem prieskumu jednotlivých editorov a správcov súborov. Popisujem pre nich porovnania, snažím sa čo najlepšie ukázať možnosti a využiteľnosť týchto komponent, a prečo som sa rozhodol práve pre tie komponenty, ktoré sú použité aj vo finálnej verzii programu.

Na konci práce sú vybrané tie najzaujímavejšie a najdôležitejšie časti programu. Ukážky kódu sú doplnené aj o stručné popisy detailov implementácie. V závere potom zhŕňam dosiahnuté výsledky.

2 Informačný systém DB Education

2.1 Pôvodný informačný systém

Pôvodný informačný systém obsahoval dve užívateľské roly, a to **študent** a **učiteľ**. Študent si mohol v informačnom systéme iba prezerať, prípadne sťahovať materiály k štúdiu. Všetky ostatné veci, ako prihlásenie na zápočet, odovzdanie projektov alebo prihlásenie na opravnú písomku si musel študent konzultovať s vyučujúcim individuálne. Učiteľ fungoval ako administrátor informačného systému, ale práca to bola zdĺhavá, a nie príliš efektívna. Bolo potrebné sa pomocou RDC vzdialene pripojiť na server, na ktorom bežal informačný systém, a takto upravovať súbory a zložky. Behom procesu mohlo vzniknúť množstvo problémov, ako napríklad, že jeden súbor upravovali súčasne dvaja užívatelia. Tým dochádzalo k neželaným situáciám, vrátane vzniku fantómových objektov, či špinavého zapisovania (pojmy dobre známe z prostredia databázových systémov). V IS taktiež chýbalo veľa potrebných a užitočných funkcií, ako napríklad vytvorenie nových ročníkov, predmetov, posielanie aktuálnych informácií, týkajúcich sa predmetov, priamo na študentské alebo učiteľské emaily, a mnoho iných.

2.2 Ciele bakalárskej práce

Cieľom bakalárskej práce bolo vytvoriť ucelený a spoľahlivo fungujúci informačný systém na správu databázových predmetov a materiálov s nimi spojenými. V informačnom systéme existujú štyri užívateľské roly: **administrátor**, **učiteľ**, **študent** a **anonymný hosť**. Každá rola má svoje vlastné práva a možnosti v rámci portálu DB Education. Od systému sa očakáva hlavne spoľahlivosť a prehľadnosť, pretože je využívaný neustále. Dôležitým bodom práce bolo vytvoriť prehľadný administratívny systém pre administrátora IS. Administrátor má preto množstvo nových funkcií, napríklad môže priamo upraviť vzhľad webovej stránky pomocou editora, spravovať súbory v rámci IS, posilať novinky ohľadom predmetov priamo na študentské/učiteľské emaily, a veľa iných. Systém funguje ako hlavná informačná stránka pre študentov a vyučujúcich databázových predmetov. Študenti si môžu prezerať alebo sťahovať materiály k štúdiu, môžu priamo vkladať vypracované projekty do IS a taktiež sa prihlasovať na zápočty. Na druhej strane, vyučujúci majú prehľad o svojich študentoch, či odovzdali projekty v určenom čase a v správnom formáte. V neposlednej rade, materiály k štúdiu si môžu prezerať aj užívatelia, ktorí nie sú súčasťou univerzity, na toto slúži rola guest.

3 Analýza

3.1 Datová analýza

3.1.1 Vstupy do systému

Pred samotnou implementáciou sme rozanalyzovali celkovú funkcionálnosť a rozvrhli si jednotlivé časti systému. Rozhodli sme sa, že vytvoríme štruktúru systému, ktorá bude rozdeľovať systém na časti. Zistili sme, že bude treba ukladať informácie o jednotlivých užívateľoch, predmetoch, projektoch, termínoch, atď. Na základe týchto zistení sme vytvorili akýsi zoznam, ktorý obsahoval nasledovné časti: *User*, *Subject*, *Course*, *EmailList*, *Project*, *ProjectUpload*, *Schedule*, *ScheduleStudent* a *AcademicYear*. Z tohoto zoznamu neskor vznikol aj výsledný datový model. Nižšie popisujem rozdelenie uložených dát.

User slúži na uloženie informácií o užívateľoch, a ich rolách. Dôležité je poznamenať, že sa tu ukladajú iba užívatelia, ktorým je pridelená rola Admin alebo Teacher. Ostatní užívatelia majú defaultnú užívateľskú rolu Student. Taktiež sa tu ukladá údaj, či je povolené prihlásenie do IS v roli Guest.

Asi najdôležitejším a taktiež najťažším rozdelením ukládaných dát bol zoznam *Subject*. Sú tu uložené údaje o tom, v ktorom ročníku (akademickom roku) sa vyučuje aký predmet, fyzická cesta k xml súboru, ktorý uchováva HTML stránku, a či sa jedná o predmet vyučovaný v českom alebo anglickom jazyku, a či ide o predmet určený pre bakalárske alebo magisterské štúdium. Vo výslednom datovom modeli je nakoniec aj ukázané, ako je tabuľka *Subject* dôležitá pre celkové fungovanie IS (sú na ňu naviazané skoro všetky tabuľky).

V zozname *Course* sa uchováva informácia o každom jedinečnom predmete, teda jeho názve a skratke. Predmet uložený v tomto zozname je jedinečný pre celý informačný systém, ale môže byť použitý v rámci jednotlivých ročníkov.

EmailList uchováva informácie o tom, ktorý užívateľ je prihlásený na odber noviniek v rámci akého predmetu.

Project uchováva informácie o jednotlivých projektoch, a to názov projektu, termín odovzdania, akého predmetu sa projekt týka a v poslednej rade aj špecifikáciu daného projektu.

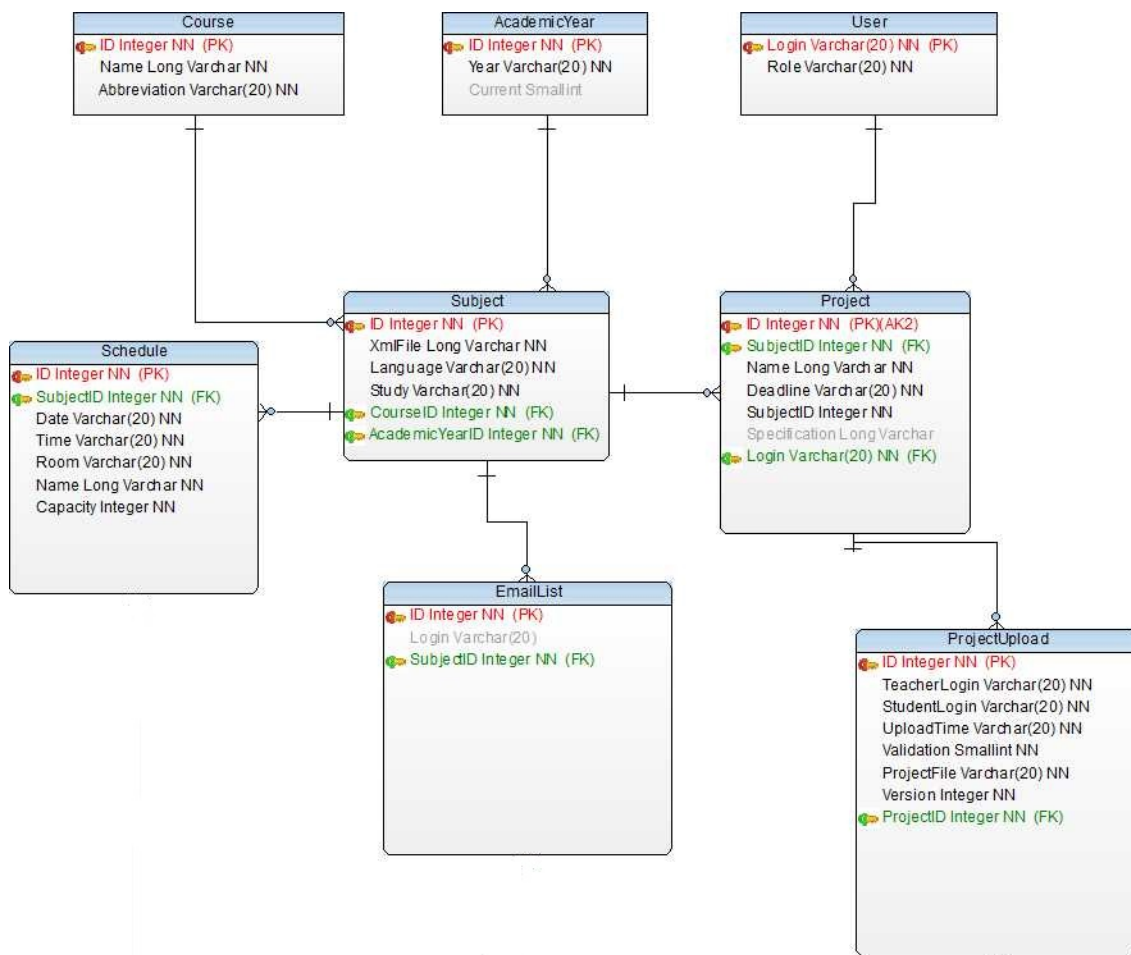
V zozname *ProjectUpload* sú uložené dáta o jednotlivých uploadovaných projektoch. Sú tu uložené informácie o študentovi, ktorý projekt odovzdal, pre ktorého vyučujúceho je projekt určený. Ďalej sú tu informácie kedy bol projekt odovzdaný, či má projekt správny tvar a či obsahuje správne súbory, ale aj informáciu o verzii projektu a fyzickú cestu k odovzdaným súborom. Nakoniec je tu aj informácia, o ktorý projekt vlastne ide.

Schedule uchováva dáta o termínoch zápočtových testov. Sú tu uložené informácie ako názov zápočtového testu, pre aký predmet je zápočtový test určený. Ďalej sú tu uložené klasické informácie ako kapacita, čas a miesto konania.

ScheduleStudent slúži iba na uloženie informácie, ktorý študent je prihlásený na ktorý zápočtový test.

AcademicYear ukladá informácie o ročníkoch (akademických rokoch). Je tu uložený názov ročníka, a ktorý ročník je práve aktuálny.

3.1.2 Datový model



Obr. 1: ER-Diagram systému DB Education

3.2 Užívateľské role

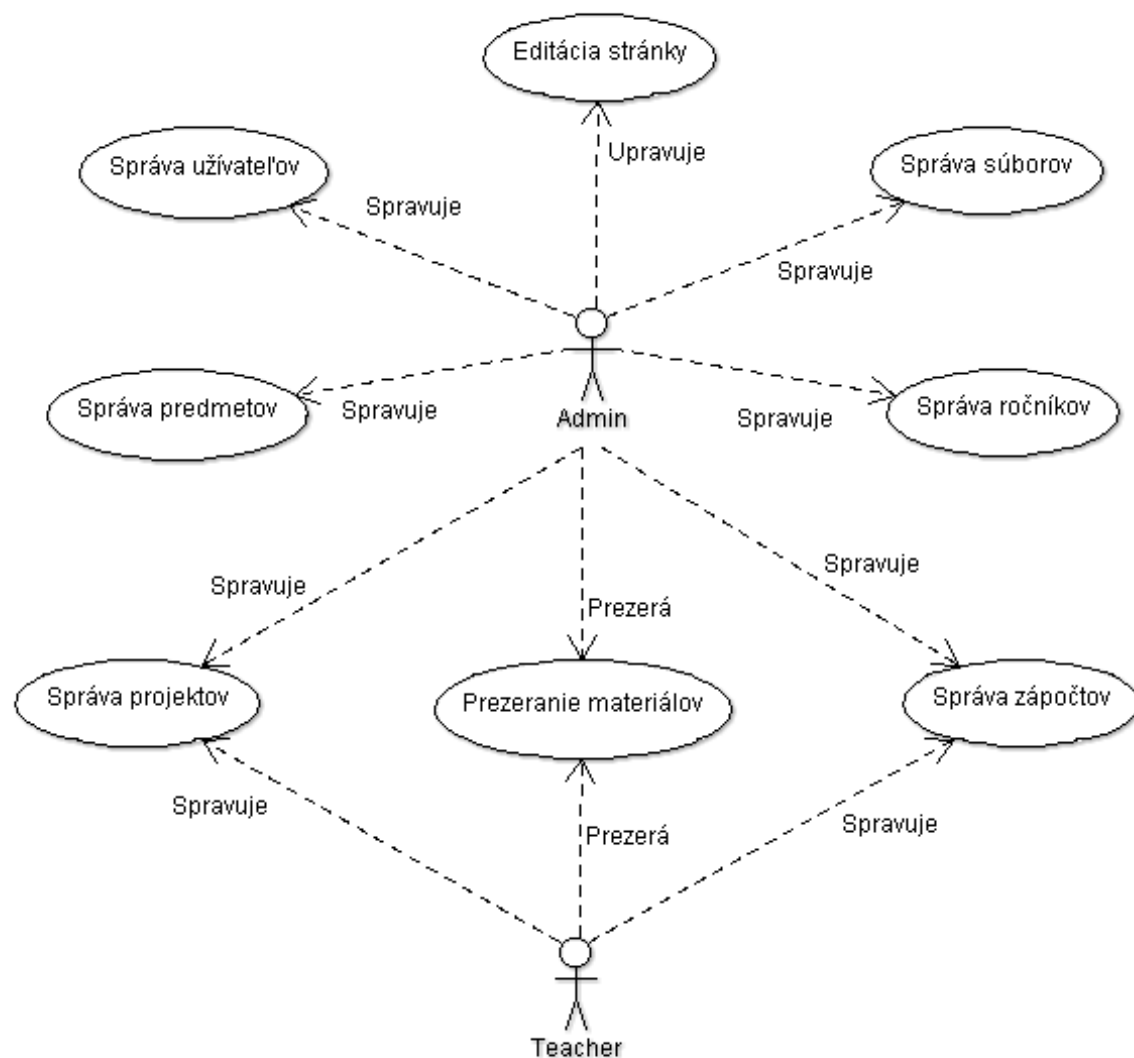
Tento informačný systém budú používať štyri skupiny ľudí:

1. **Admin** - má na starosti správu a aktualizáciu informácií v informačnom systéme. Vytvára a edituje jednotlivé roly užívateľov, vytvára/edituje informácie o zápočtových testoch, projektoch, ročníkoch a predmetoch. Môže upravovať obsah webových stránok a spravovať súbory a zložky v rámci IS. Ďalej môže aktualizovať predmety a súčasne tieto novinky posielat' na študentské/učiteľské emaily
2. **Teacher** - môže si prezerať a spravovať projekty od svojich študentov, taktiež môže vytvárať termíny zápočtových testov pre študentov
3. **Študent** - môže uploadovať projekty a prihlasovať sa na zápočty
4. **Guest** - prihlásený guest môže prechádzať stránky predmetov a zobrazovať si prístupné súbory a materiály, aj keď užívateľ nie je súčasťou univerzity

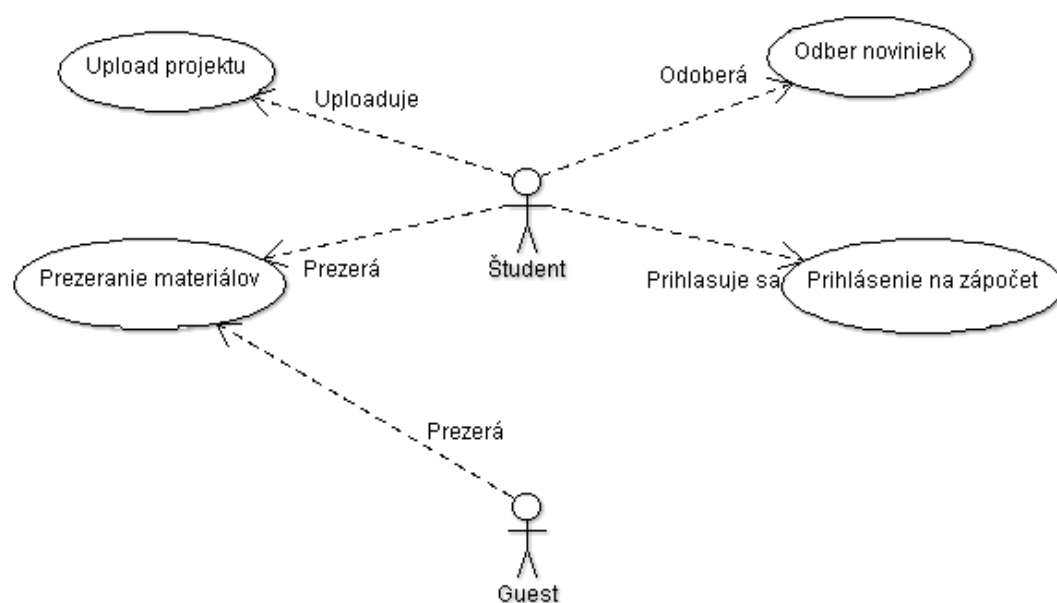
Užívatelia s rolou *Admin*, *Teacher* a *Študent* musia byť súčasťou univerzity, pretože prístupové meno a heslo do IS sa kontroluje pomocou LDAP¹ databázy.

¹jedinečné meno a heslo v rámci univerzity

3.3 Diagramy užívateľských rolí



Obr. 2: Use-case diagram pre rolu Admin a Teacher



Obr. 3: Use-case diagram pre rolu Študent a Guest

3.4 Funkčná analýza

3.4.1 Zoznam funkcií

1. Správa užívateľov

Tabuľka: User, *Zodpovednosť:* Admin

- 1.1 Vloženie nového užívateľa
- 1.2 Zmena roly užívateľa
- 1.3 Vymazanie užívateľa
- 1.4 Povolenie/zrušenie užívateľskej roly Guest

2. Správa zápočtov

Tabuľka: Schedule

- 2.1 Vytvorenie termínu zápočtového testu
Zodpovednosť: Admin, Teacher
- 2.2 Editácia termínu zápočtového testu
Zodpovednosť: Admin, Teacher
- 2.3 Vymazanie termínu zápočtového testu
Zodpovednosť: Admin, Teacher
- 2.4 Prihlásenie na termín zápočtu daného predmetu
Zodpovednosť: Študent
- 2.5 Odhlásenie z termínu zápočtu daného predmetu
Zodpovednosť: Študent

3. Správa projektov

Tabuľka: Project

- 3.1 Vytvorenie projektu
Zodpovednosť: Admin, Teacher
- 3.2 Editácia projektu
Zodpovednosť: Admin, Teacher
- 3.3 Špecifikácia zadania projektu
Zodpovednosť: Admin, Teacher
- 3.4 Vymazanie projektu
Zodpovednosť: Admin, Teacher

- 3.5 Prezeranie projektov
Zodpovednosť: Admin, Teacher
- 3.6 Upload projektu pre daný predmet *Zodpovednosť: Študent*

4. Správa ročníkov

Tabuľka: AcademicYear, Subject, Zodpovednosť: Admin

- 4.1 Vytvorenie akademického roka
- 4.2 Editácia akademického roka
- 4.3 Nastavenie aktuálneho akademického roka

5. Správa predmetov

Tabuľka: Course, Subject, Zodpovednosť: Admin

- 5.1 Vytvorenie nového predmetu
- 5.2 Vytvorenie predmetu pre daný ročník
- 5.3 Editácia predmetu v danom ročníku
- 5.4 Vymazanie predmetu z daného ročníka

6. Správa stránky predmetu

Tabuľka: Course, EmailList

- 6.1 Editácia stránky predmetu
Zodpovednosť: Admin
- 6.2 Aktualizácia predmetu a zasielanie noviniek študentom a učiteľom na email
Zodpovednosť: Admin
- 6.3 Správa súborov v rámci IS
Zodpovednosť: Admin
- 6.4 Prihlásenie na odber noviniek daného predmetu
Zodpovednosť: Teacher, Študent
- 6.5 Odhlásenie z odberu noviniek daného predmetu
Zodpovednosť: Teacher, Študent
- 6.6 Prezeranie materiálov dostupných na portáli DB
Zodpovednosť: Admin, Teacher, Študent, Guest

7. Prihlásenie do IS

Tabuľka: User Zodpovednosť: Admin, Teacher, Študent, Guest

- 7.1 Prihlásenie do IS s danými právami

3.5 Detailný popis funkcií

3.5.1 Vytvorenie predmetu pre daný ročník

Funkcia 5.2

Aktér: Admin

Požiadavok: Scenár Prihlásenie admina prebehol úspešne

Scenár:

1. Admin vyberie z dropdown listu predmet, ktorý chce vložiť do akademického roka
2. Systém automaticky zobrazí skratku daného predmetu
3. Admin ďalej vyplní údaje, či sa jedná o predmet vyučovaný v českom alebo anglickom jazyku a či ide o predmet pre bakalárske alebo magisterské štúdium
4. Admin si ďalej vyberie, či chce predmet uložiť ako prázdnu stránku, prázdnu stránku s pred vytvorenou tabuľkou alebo bude stránka totožná so stránkou z predchádzajúceho ročníka
5. Admin klikne na tlačítko Vložiť predmet, čím sa predmet pridá do zoznamu predmetov pre daný ročník

Alternatívny scenár:

- 1.1 Predmet sa v zozname predmetov nenachádza
- 1.2 Admin klikne na odkaz, ktorý sa nachádza pod dropdown listom, ktorý ho presmeruje na vytvorenie nového predmetu
- 1.3 Admin vyplní údaje o názve predmetu a skratke predmetu
- 1.4 Admin klikne na tlačítko Pridať, ktoré novovytvorený predmet vloží medzi ostatné predmety
1. ...

3.5.2 Vytvorenie termínu zápočtového testu

Funkcia 2.1

Aktér: Admin, Teacher

Požiadavok: Scenár Prihlásenie admina/teachera prebehol úspešne

Scenár:

1. Admin/Teacher vyberie predmet z dropdown listu, pre ktorý chce vytvoriť termín zápočtového testu
2. Admin/Teacher vyplní údaje potrebné k vytvoreniu termínu zápočtového testu
3. Admin/Teacher klikne na tlačítko Pridať, ktoré pridá termín zápočtového testu do zoznamu termínov

Alternatívny scenár:

- 2.1 Admin/Teacher nevyplnil všetky údaje
2. ...

3.5.3 Upload projektu

Aktér: Študent

Požiadavok: Scenár Prihlásenie študenta prebehol úspešne

Scenár:

1. Študent vyberie z dropdown listu učiteľa, predmet a projekt
2. Študent klikne na tlačítko Vybrat soubor
3. Študent vyberie z počítača súbor, ktorý chce uploadovať
4. Študent klikne na tlačítko Upload

Alternatívny scenár:

- 4.1 Upload projektu nebol úspešný (chyba serveru, strata internetového spojenia)

4 Použité technológie

4.1 ASP.NET

Pôvodne boli všetky webové stránky čisto statické, to znamená, že server poslal odpoveď klientovi, tak ako ju mal uloženú. Statické webové stránky majú veľa obmedzení, preto začal vznikať web dynamický. Jednou možnosťou implementácie dynamického webu je práve využitie technológie **ASP.NET**[12][19].

ASP.NET je webový framework, ktorý slúži na tvorbu dynamických webových stránok. Obsahuje množstvo knižníc, pomocou ktorých je možné riešiť všeobecné problémy (práca s formulármi, autentifikácia, práca s databázou, a iné). ASP.NET je výbornou voľbou na vytvorenie webových portálov, ale hodí sa aj na vytvorenie menších projektov. Microsoft ponúka dva hlavné spôsoby, ako vytvárať webové stránky: MVC a WebForms. Obidva spôsoby sú zaujímavé a obidva sa pozerajú na webovú aplikáciu trochu inak.

4.1.1 MVC

Rozdeľuje aplikáciu na tri vrstvy: Model, View a Controller, kde každá časť má svoju úlohu. Model obsahuje celú logiku programu, pracuje s funkciami a s databázou. View je grafická časť webu, teda to čo užívateľ vidí. Controller je spojenie medzi vrstvou Model a vrstvou View, prijíma dáta od užívateľa a pošle ich vrstve Model na spracovanie.

4.1.2 WebForms

Funguje ako WinForms, teda rozdelenie stránky na okná, tlačítka, popisky, atď a týmto ovládacím prvkom (Control) priradzuje funkcionality. Špecifikom WebForms je, že sa snažia priniesť "stavovosť" do bezstavového webu. Na klasickom webe sa po odoslaní formulára všetky informácie stratia, preto WebForms využíva ViewState. ViewState je skryté formulárové okno, ktoré na výslednej HTML stránke uchováva informácie a posielajú ich serveru spolu s požiadavkou. WebForms je starší koncept ako MVC, ale stále je veľmi využívaný. Tento koncept je použitý aj v tomto informačnom systéme.

4.2 SQLite

Súčasťou informačného systému DB Education musí byť logicky aj nejaké úložisko pre dáta. V tomto prípade sme sa rozhodli využiť databázu **SQLite**[11][15]. Pred použitím SQLite nie je nutná inštalácia, a dokonca ani žiadna konfigurácia. Táto databáza sa ukladá do jediného súboru, ktorý môže byť uložený hocikde v stromovej štruktúre súborov. SQLite má ale jednu veľkú nevýhodu oproti väčšine iných SQL databáz, a to, že

nedokáže pracovať na zdieľanom úložisku, tiež nedokáže podporovať veľké množstvo súčasných operácií a pripojení. Napriek týmto nevýhodám to v našom prípade nevadí, pretože sa jedná iba o malú lokálnu databázu.

4.3 Pôvodný editor

Jedným zo zadaných bodov bakalárskej práce bolo vytvoriť editor na úpravu webových stránok. Hoci portál DB Education umožňoval editáciu webových stránok už predtým, spôsob, akým sa k úpravám pristupovalo nebol ideálny. Pôvodná verzia úpravy stránok bola postavená na XML súboroch, čiže nebolo možné upravovať ich priamo na webe. K editácii sa pristupovalo cez jednoduchý TextBox, v ktorom sa menila HTML štruktúra stránky. Nutnými predpokladmi pre editáciu bola znalosť HTML a XML syntaxe. Práve kvôli tomuto bola správa a úprava stránky časovo náročná a mohlo dochádzať k syntaktickým chybám. Oveľa lepšou voľbou by bolo preto použiť WYSIWYG editor, kde by užívateľ stránku upravoval priamo (aj bez znalosti HTML štruktúry).

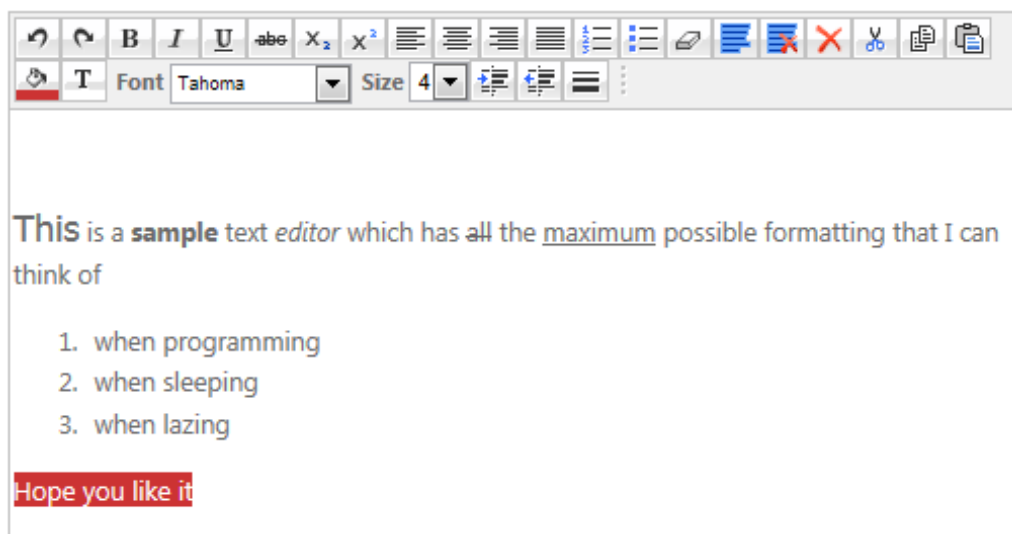
4.4 HTML/XML editory

4.4.1 Ajax Control Toolkit

Ajax Control Toolkit[7] je open source projekt, postavený na Ajax ASP.NET frameworku. Jedná sa o spoločnú snahu medzi Microsoftom a ASP.NET Ajax komunitou vytvoriť unikátny nástroj na písanie použiteľného, prispôsobiteľného a rozširiteľného ASP.NET Ajax frameworku.

Výhody:

Ajax Control Toolkit obsahuje bohatú sadu ovládacích prvkov, ktoré môže užívateľ použiť k vytvoreniu vysoko citlivej a interaktívnej webovej aplikácie. Ajax Control Toolkit obsahuje viac ako 40 ovládacích prvkov ako napr. automatické dopĺňovanie textu, výber farieb, HTML editor a pod. Ajax Control Toolkit-u umožňuje vytvoriť ASP.NET webovú aplikáciu s použitím "drag & drop" z Visual Studia priamo do ASP.NET Web Forms stránky.



Obr. 4: Vzhľad komponenty Ajax Control Toolkit

Nevýhody:

Webová stránka sama o sebe je bezstavová a po znovunačítaní stránky, si nepamäta predošlé údaje. Keď je stránka písaná v ASP.NET, nespolupracuje s AJAX-om správne. Ajax neposiela ako odpoveď zo serveru iba vyžiadané informácie, ale pošle dáta z celej stránky. To jednak zaťažuje server a klienta, tiež spomaľuje rýchlosť webu. Taktiež je to bezpečnostné riziko, lebo v odpovedi sa posielajú aj citlivé dáta, ako napr. heslá [18]

Druhým problémom je správanie Control Toolkitu po zmene a následnom uložení dokumentu. Vznikajú napríklad chyby v HTML syntaxi, Toolkit samovoľne pridáva tagy <div> alebo automaticky vytvára medzery a odstavce aj tam kde to užívateľ nepotrebuje. Vytváranie zle uzavretých a navzájom vnorených tagov, prázdnych značiek ako aj zmeny formátovania textu zničili takmer každú stránku aj po niekoľko málo pokusoch o editáciu.

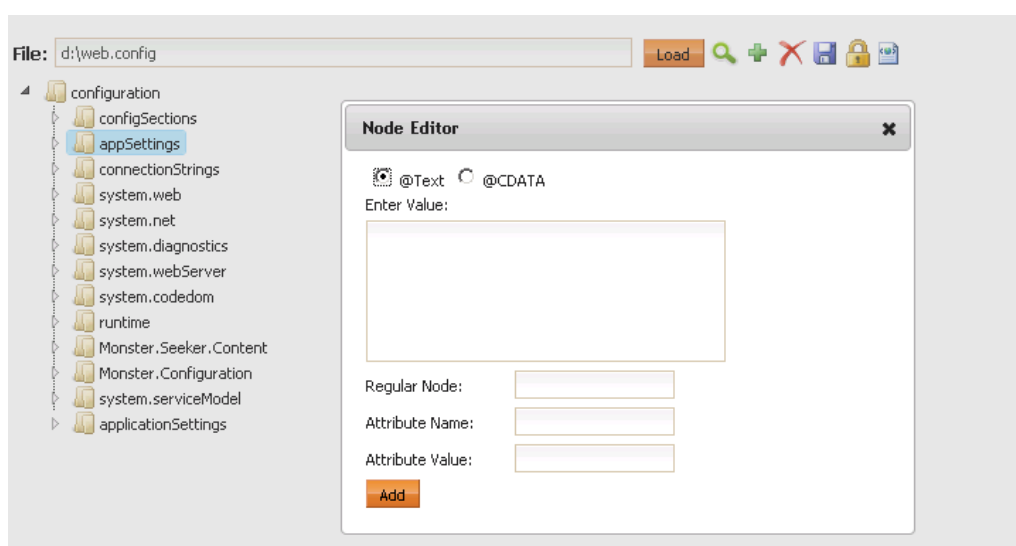
4.4.2 XML Webpad Editor

XML Webpad Editor[5] je framework, slúžiaci na prezeranie a editovanie XML súborov, ktoré sú súčasťou ľubovoľnej ASP.NET webovej aplikácie. Editor je postavený na technológii C#, .NET 3.5 a JavaScript. Má vstavaný webový GUI XML editor, kde implementuje stromovú štruktúru dokumentu, a taktiež JS DOM mapovací algoritmus. XML Webpad Editor vytvoril Udayan Das a distribuuje ho pomocou **CodePlex**. CodePlex[4] je open source webhostingový projekt, na ktorom môže pracovať tak povediac hocikto, kto

chce nejakým spôsobom pomôcť vo vývoji softwaru. Užívatelia môžu vytvárať vlastné projekty, alebo spolupracovať a aktívne sa podieľať na vývoji projektu iného užívateľa.

Výhody:

Narozdiel od iných nástrojov a editorov, Webpad v jeho jadre nie je len samostatná webová aplikácia, ale skôr poskytuje framework (XMLWebPad.dll a xmlwebpad.js) na vytvorenie vlastného prehliadača a editora s možnosťou integrácie do ASP.NET aplikácie. XML Webpad nie je ASP.NET serverová komponenta (server control), vďaka tomu poskytuje plnú kontrolu nad užívateľským rozhraním. Editor nielenže dovoľuje modifikáciu dát, ale dovoľuje aj meniť XML štruktúru.



Obr. 5: Vzhľad komponenty XML WebPad Editor

Nevýhody:

XML WebPad Editor má aj svoje nevýhody, a to nemožnosť priamo upravovať štruktúru a vzhľad HTML stránky. Editor je veľmi neprehľadný, pretože to nie je WYSIWYG editor, a úprava stránky je takpovediac nemožná. Ďalšou nevýhodou je zložité vkladanie a úprava dát. Každý záznam je treba vložiť samostatne, čo je pre naše potreby zdĺhavý proces. Editor je stále vo vývoji a preto obsahuje aj ďalšie drobné chyby, ktoré bude treba v budúcnosti odstrániť.

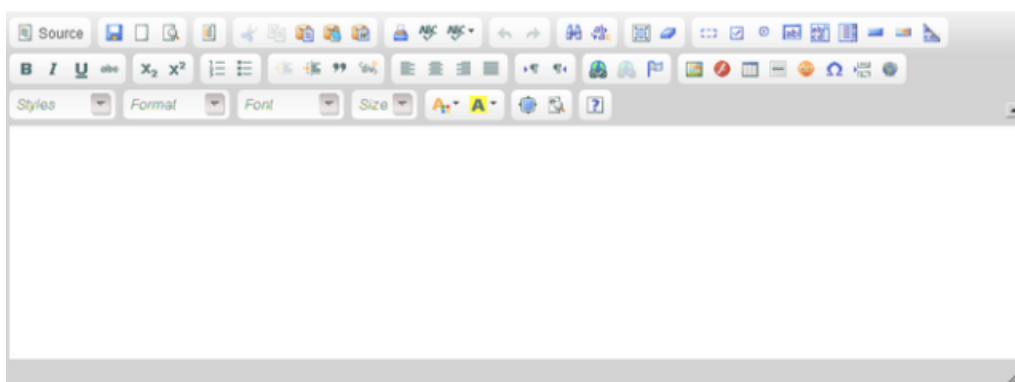
4.4.3 CKEditor

CKEditor[1] je WYSIWYG HTML textový editor navrhnutý tak, aby čo najviac zjednodušil tvorbu webového obsahu. Vyvíjaný je spoločnosťou CKSource. CKEditor je open source aplikácia, čo znamená, že môže byť upravený spôsobom, akým najlepšie vyhovuje požiadavkám programátora. Práve toto ho robí veľmi atraktívnym v očiach komunity, ktorá ho rozširuje a vylepšuje.

Prvá verzia CKEditora (pred tým bol pomenovaný FCKeditor) bola vydaná v marci roku 2003. Prvá verzia editora podporovala XHTML a Internet Explorer 5.0. V ďalších dvoch rokoch sa o editor začali zaujímať známe spoločnosti, napríklad Oracle a IBM, ktoré začali editor aj podporovať. V tej dobe začal byť editor veľmi žiadaný, a po prekročení hranice troch miliónov stiahnutí bol vylepšený do verzie CKEditor 3.0. Nová verzia mala optimalizovaný výkon, prístup, použiteľnosť, ale hlavne nové užívateľské rozhranie. V decembri 2012 bola vydaná aktuálna verzia 4.0. V editore bol aktualizovaný zdrojový kód, boli pridané nové funkcie a doplnky[2].

Výhody:

CKEditor je preložený do viac ako 60 svetových jazykov, preto dokáže automaticky detekovať jazyk a lokalizovať rozhranie podľa užívateľa. Editor podporuje písanie a editovanie stránok sprava doľava, preto môžu byť stránky písané napr. v arabskom alebo hebrejskom jazyku. Vývojári kladú vysoký dôraz na plynulú prácu s editorom, práve toto dostáva CKEditor pred svoju konkurenciu. Nezáleží koľko editorov je v stránke implementovaných, vždy sa načítajú rýchlo



Obr. 6: Vzhľad komponenty CKEditor

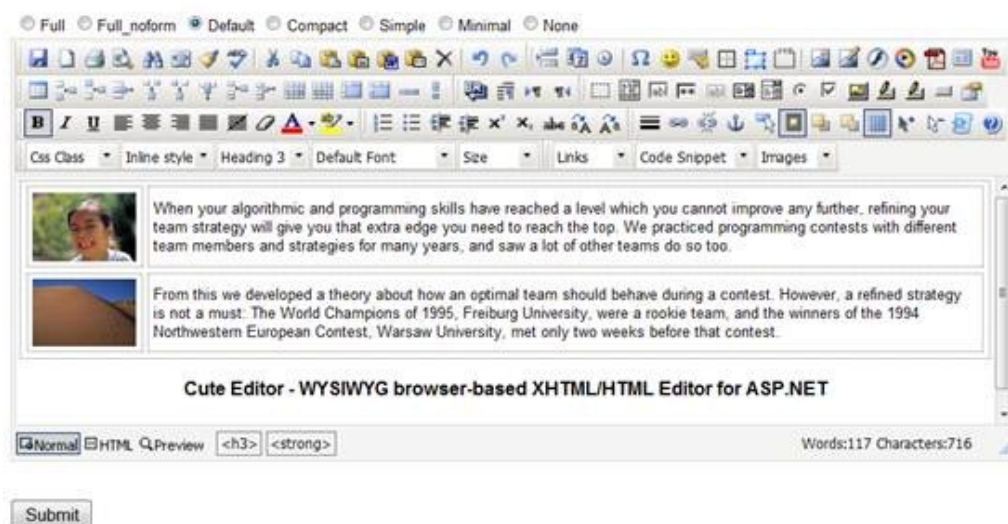
V porovnaní s ostatnými editormi neboli nájdené žiadne nedostatky CKEditora. Toto tvrdenie podporuje aj prieskum na stránke SocialCompare ².

4.4.4 CuteEditor/RichEditor

CuteEditor/RichEditor[6] je editor vyvíjaný spoločnosťou CuteSoft. Je to veľmi rýchly, jednoduchý a často využívaný WYSIWYG HTML Editor. Okrem ASP.NET webových aplikácií, funguje taktiež s PHP stránkami. Môže slúžiť ako súčasť redakčného systému pre užívateľov webu.

Výhody:

CuteEditor funguje na všetkých dostupných prehliadačoch a operačných systémoch. Je dlhodobo optimalizovaný a snaží sa vytvoriť prostredie podobné tomu z Microsoft Word pre jednoduchšie používanie. Implementácia editoru do web stránky je veľmi jednoduchá. Podporuje najnovšie webové štandarty, ako HTML5, tak aj XHTML 2.0. CuteEditor dokáže importovať súbory z Word-u a RTF priamo do HTML bez straty formátovania. Umožňuje taktiež ukladať obrázky do pripojenej SQL databázy. Administrátor stránky môže nastavovať rôzne úrovne prístupu pre rôznych užívateľov v rámci CuteEditora. Taktiež, ponúka pokročilé možnosti uploadu[13].



Obr. 7: Vzhľad komponenty CuteEditor/RichEditor

²<http://socialcompare.com/en/comparison/javascript-online-rich-text-editors>

Nevýhody:

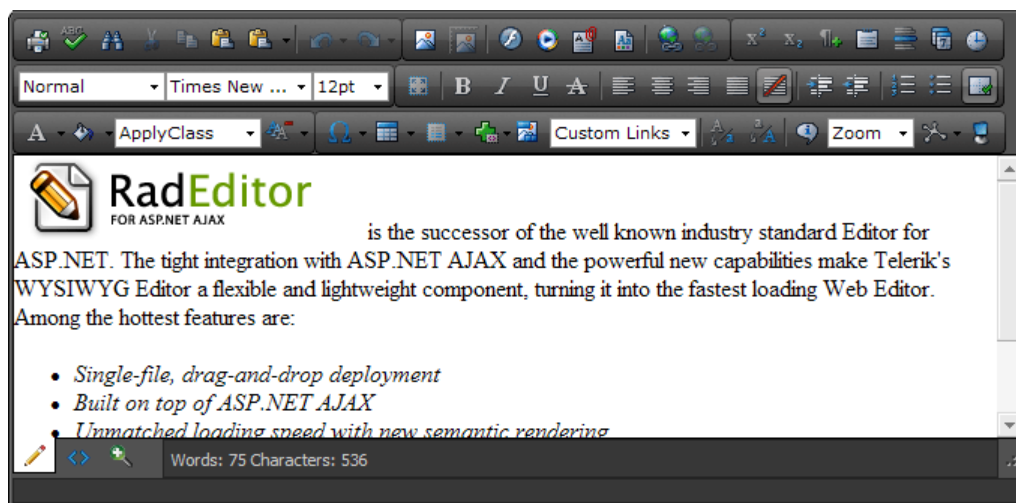
CuteEditor trpí niekoľkými nedostatkami. Najzávažnejším je nemožnosť editovať tabuľky. Editor obsahuje aj množstvo skrytých chýb. Dokumentácia je veľmi biedna a podľa názorov iných užívateľov je online podpora na veľmi nízkej úrovni.

4.4.5 RadEditor

RadEditor[16] je ďalší ASP.NET editor pre HTML, vyvíjaný spoločnosťou Telerik. Umožňuje prácu podobnú s prostredím Microsoft Word. Dokáže editovať text, tabuľky, hyperlinkové odkazy a množstvo ďalších častí web stránky. Umožňuje implementovať vlastné CSS pre zmenu vzhľadu a rozšírenie funkcionality. Obsahuje integrovaných správcov odkazov, obrázkov, videí, hudby a ďalších typov súborov.

Výhody:

RadEditor podporuje kontrolu gramatiky. Dokáže konvertovať web do rôznych štandardov HTML a XHTML. Stránka môže byť editovaná prostredníctvom WYSIWYG editora, alebo priamo, úpravou zdrojového kódu. Na rozdiel od iných editorov, RadEditor podporuje úpravu kompletnej web stránky, vrátane tagov ako <head>, <body>, <meta>, atď. Spoločnosť Telerik poskytuje rozsiahle klientské a serverové API a množstvo ďalších komponent, ktoré spolupracujú s RadEditorom.



Obr. 8: Vzhľad komponenty RadEditor

Nevýhody:

Podobne ako CKeditor, aj RadEditor je veľmi pokročilý nástroj. Neobsahuje takmer žiadne chyby, či negatíva, ale je to nepriamy nástupca AJAX Control Toolkit Editor. Z tohto dôvodu editor zdedil aj AJAX komunikáciu a s ASP.NET AJAX implementáciou spojené problémy, spomínané v kapitole 4.4.1.

4.5 Správcovia súborov

Ďalším základným bodom v zadaní bakalárskej práce bolo nasadenie súborového systému, pre logické prepojenie s HTML editorom. Pôvodne portál DB Education neumožňoval žiadnym spôsobom prácu so súbormi a priečkami. Administrátor sa musel pripojiť priamo na server pomocou vzdialenej plochy a natvrdo upravovať súbory a zložky. Tento proces bol veľmi zdĺhavý, komplikovaný a nekomfortný. Navyše ohrozoval stabilitu serveru. Našťastie, aj v tomto prípade existuje niekoľko elegantných a sofistikovaných riešení v podobe online správco súborov. Rovnako ako HTML/XML editor, správca súborov by mal spĺňať požiadavky ako napríklad open-source prístup, jednoduchosť používania, efektivitu práce a iné. Po ďalšom prieskume sme našli editory, ktoré spĺňali požiadavky na nasadenie:

- HTTP Commander
- CKFinder
- ELFinder

4.5.1 HTTP Commander

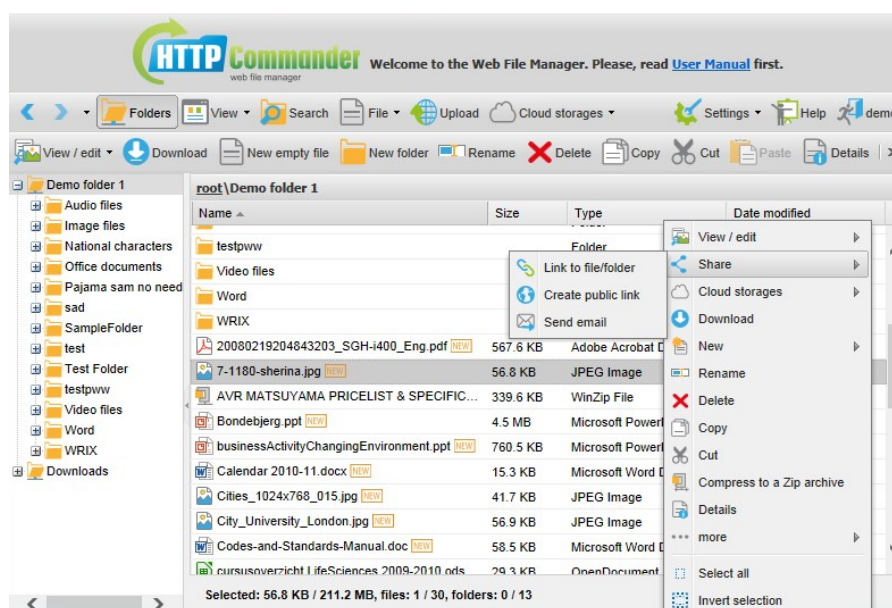
HTTP Commander[8] je webový správca súborov a priečinkov, používaný na Windows serveroch. Bol vytvorený firmou Element IT, prioritne určený pre väčšie spoločnosti, ktoré potrebujú prehľadný manažment súborov.

Výhody:

Komponenta slúži hlavne na zdieľanie a manažment súborov - preto je to najlepšia alternatíva k FTP³, WebDav[20] alebo Sharepoint[14]. Užívatelia môžu pracovať na

³protokol slúžiaci na prenos súborov medzi počítačmi za použitia počítačovej siete

diaľku s osobnými alebo zdieľanými zložkami a upravovať dokumenty on-line uložené na serveri. HTTP Commander podporuje všetky známe webové prehliadače. Podporuje formuláre a samozrejme aj Windows autentifikáciu. Inštalácia a konfigurácia je hotová za pár minút, kvôli tomu, že nepotrebuje SQL server. Podporuje viac ako 300 formátov zdieľaných súborov a umožňuje ich otvárať a upravovať priamo na webe. Umožňuje integráciu s GoogleDrive alebo Dropbox - užívatelia si môžu súbory sťahovať priamo na svoje úložisko.



Obr. 9: Vzhľad komponenty HttpCommander

Nevýhody:

Podľa prieskumu a recenzií od zákazníkov má aj HTTP Commander určité nedostatky. Má slabšiu zákaznícku podporu od výrobcu, a v prípade väčších modifikácií je implementácia zložitá. HTTP Commander taktiež nie je open source a teda nespĺňa naše požiadavky.

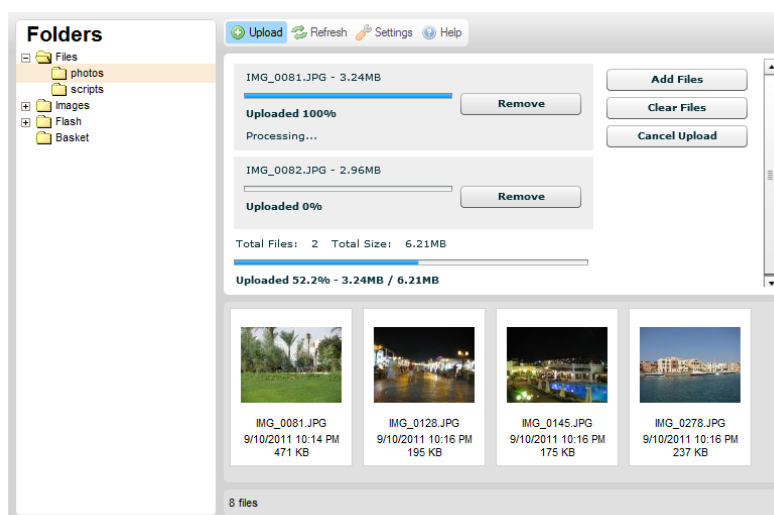
4.5.2 CKFinder

CKFinder[3] je webový správca súborov, postavený na AJAX technológii. Tento správca súborov bol vytvorený firmou CKSource, a je to jeden z najpoužívanejších správčov sú-

borov vôbec. Hovorí o tom už samotný fakt, že ho využívajú spoločnosti ako IBM, Oracle, Adobe a iné.

Výhody:

CKFinder obsahuje náhľad do stromovej štruktúry priečinkov a súborov, vďaka čomu umožňuje prehľadnú správu a manažment. CKFinder podporuje všetky bežne využívané webové prehliadače. Kontextové menu sa automaticky prispôsobuje vybranému súboru alebo priečinku, čo veľmi zjednodušuje a urýchľuje prácu. Súborový správca obsahuje všetky dôležité informácie o vybranom súbore alebo priečinku, čo mimoriadne zjednodušuje prehľadnosť manažmentu. CKFinder umožňuje integráciu a spoluprácu s editorom CKEditor. Obsahuje množstvo ďalších užitočných nástrojov (podpora drag-and drop, podpora viacnásobného uploadu, atď.), ako aj plnú kontrolu nad zdrojovým kódom.



Obr. 10: Vzhľad komponenty CKFinder

Nevýhody:

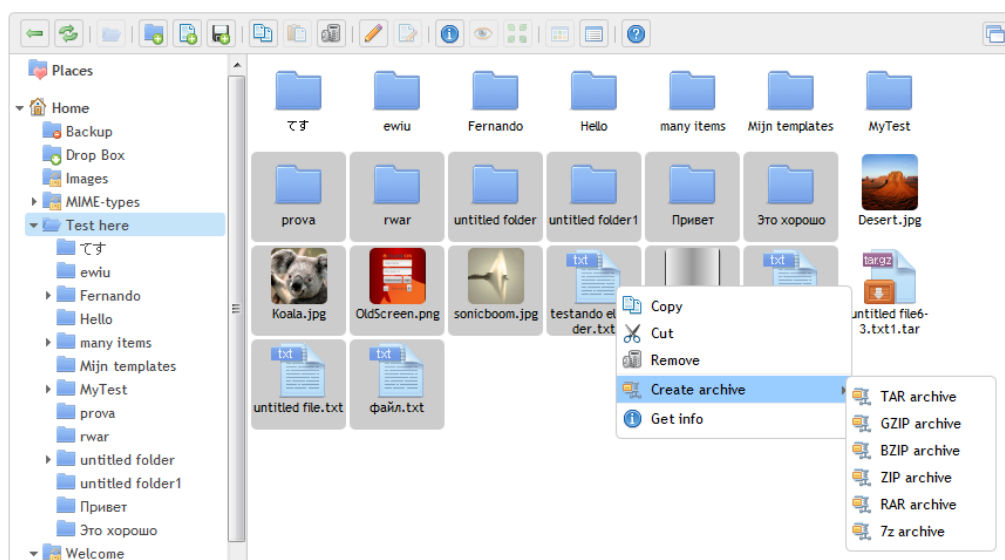
Ako jeden z možných nedostatkov by som videl fakt, že stále obsahuje určité bezpečnostné problémy. Dobrou správou je, že výrobca pracuje na odstránení týchto chýb a snaží sa túto komponentu stále vylepšovať. CKFinder nie je open source komponenta, a tak nevyhovuje našim minimálnym požiadavkám na správcu súborov.

4.5.3 ELFinder

ELFinder[10] je open-source správca súborov, napísaný v JavaScripte použitím jQuery. Serverová časť komponenty je implementovaná v PHP, ASP.NET aj Pythone. Aplikácia je inšpirovaná jednoduchosťou a pohodlnosťou programu Finder, používanom na operačných systémoch MAC.

Výhody:

ELFinder výborne funguje v spojení s webovými editormi ako napr. CKEditor, elRTE[9], TinyMCE[17]. Všetky operácie sa spracovávajú na vzdialenom serveri - kopírovanie, vytváranie/editovanie zložiek a súborov, upload, atď. Táto komponenta umožňuje vysokú rozšíriteľnosť a flexibilitu konfigurácie prístupových práv, dátových typov súborov, užívateľského prostredia, atď. Umožňuje vytvorenie lokálneho súborového systému a SQL databázy. Ďalšou výhodou je jeho open source využitie.



Obr. 11: Vzhľad komponenty ELFinder

4.6 Zhrnutie a výber komponent

CuteEditor a rovnako aj RadEditor sú vynikajúce nástroje, ale nanešťastie obsahujú drobné nedostatky a chyby, ktoré nespĺňali naše požiadavky. Nakoniec sme sa rozhodli využiť editor CKEditor, ktorý má veľmi dobré referencie a je jeden z najvyužívanejších open-source editorov.

HTTP Commander síce spĺňa naše požiadavky na funkcionality, nanešťastie má pár chýb v oblasti bezpečnosti, a nie je open source. CKFinder by bol výbornou voľbou, nie len kvôli hladkej spolupráci s CKEditorom, ale aj kvôli obsahu a možnostiam, čo ponúka. Nanešťastie tento správca súborov nefunguje ako open source a je platený. Kvôli tejto nepríjemnosti sme siahli po správcovi súborov s názvom ELFinder, ktorý umožňuje jednoducho spravovať súbory a priečinky.

Po vyhodnotení prieskumu a zvážení všetkých výhod a nevýhod sme sa s vedúcim bakalárskej práce zhodli na tom, že CKEditor spolu s ELFinderom bude tá správna voľba pre administráciu portálu DB Education.

5 Implementácia

5.1 Webové rozhranie

Database Courses@cs.vsb.cz
Databázové předměty@cs.vsb.cz

www.vsb.cz www.fei.vsb.cz
www.cs.vsb.cz db.cs.vsb.cz

Úvod do databázových systémů

Last update: 24.01.2015 @ 19:20

Upravit stránku Správa souborů

ODESLAT

V případě jakýchkoli dotazů k předmětu použijte email na přednášejícího radim.baca@vsb.cz.

Aktuálně

- (13.10.) Vzorové zadání naleznete [zde](#).
- (10.10.) Popis datového modelu, jenž bude na první testu, je k dispozici [zde](#) (skript naleznete [zde](#)).
- (29.9.) Vzhledem k tomu, že se většina cvičení koná na učebnách, které nemají potřebný počet počítačů vzhledem ke kapacitě cvičení, tak bychom Vás chtěli požádat, aby jste si na cvičení nosili vlastní přenosné počítače s nainstalovaným SSMS. Děkujeme za pochopení

Software

V rámci tohoto předmětu máte vytvořenu databázi na školním SQL Server 2012. Pokyny pro připojení je možné nalézt [zde](#). Jedním s klientů (a doporučeným) je oficiální SQL Server Management Studio (SSMS). Klienta stáhnete na stránkách [elms.cs.vsb.cz](#) pod názvem "Microsoft SQL Server 2012 Express Management Studio with Service Pack 1 32/64-bit (English)". Klient je součástí i dalších instalací, ale tuto doporučujeme. Po zahájení instalace můžete nechat všechny implicitní volby a instalaci dokončit. Pro snížení velikosti instalace je možné nechat zaškrtnuty pouze volby "Client Tools Connectivity" a "Management Tools - Basic".

Podmínky udělení klasifikovaného zápočtu

Zápočet se skládá z dvou bodovaných úkolů, které se budou konat přímo na cvičeních. První test bude zaměřen na psaní SELECT dotazů (40/20). Druhý test bude zaměřen na návrh databáze a její vytvoření (20/10). Při testech bude možné pracovat se všemi materiály. U obou testů budou možné dvě opravy.

Login

Logged as:
dir0012 (admin)

Sign Out

Sekce administrátor

- ☐ Specifikace projektu
- ☐ Správa ročníků
- ☐ Správa uživatelů
- ☐ Správa zápočtů
- ☐ Projekty od studentů

České kurzy

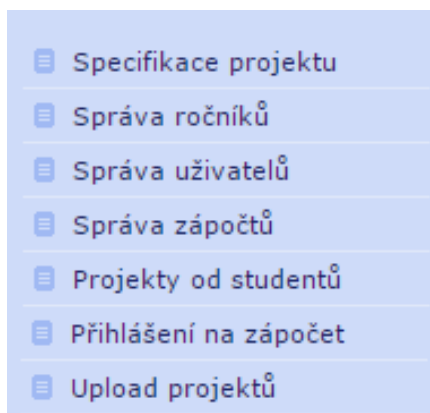
- ☐ Databázové a informační systémy 2
- ☐ Databázové systémy
- ☒ Úvod do databázových systémů

English Courses

Obr. 12: Informačný systém DB Education

Na obrázku 12 je vidno rozloženie webovej stránky na portáli DB Education. V hornej časti stránky sú nástroje na správu webovej stránky a posielanie noviniek. Vpravo je zobrazený login užívateľa a jeho aktuálna rola v systéme. Menu ďalej obsahuje sekciu pre danú užívateľskú rolu, zvyšok menu je spoločný pre všetky roly.

5.1.1 Menu



Obr. 13: Menu DB Education

Na obrázku 13 sú zobrazené všetky funkcie z menu portálu DB Education. Samozrejme každá užívateľská rola nevyužíva všetky tieto položky. Nižšie preto popisujem, kto využíva ktoré funkcie, a aká funkcionalita sa za týmito položkami z menu ukrýva.

Užívateľ s rolou **Admin** využíva nasledovné webové stránky:

- *Správa projektu* - administrátor môže vytvárať, editovať, alebo vymazávať jednotlivé špecifikácie projektov pre daný predmet
- *Správa ročníků* - administrátor môže vytvoriť nový ročník (akademický rok), pridávať predmety, ktoré sa budú v danom ročníku vyučovať, ale môže nastaviť ročník aj ako aktuálny. Cez toto menu sa môže administrátor dostať aj na ďalšiu stránku, a tou je *Správa predmětů* - na tejto stránke môže administrátor vytvárať nové kurzy univerzity, priamo vytvárať základ webovej stránky predmetu, ale samozrejme môže predmety aj editovať, či vymazávať
- *Správa uživatelů* - administrátor môže pridať nového užívateľa a priradiť mu rolu Admin alebo Teacher. Rola Student je pridaná užívateľovi automaticky, ak je súčasťou univerzity (kontrola LDAP údajov). Na tejto stránke administrátor taktiež rozhoduje, či na portáli zobrazí možnosť prihlásenia ako Guest
- *Správa zápočtů* - administrátor môže vytvárať, editovať, alebo vymazávať jednotlivé termíny zápočtov pre daný predmet. Taktiež si môže zobraziť informáciu o tom, ktorí študenti sú prihlásení na aký termín

- *Projekty od studentů* - administrátor si môže zobrazit' jednotlivé informácie o odovzdaných projektoch, ako napríklad, kto odovzdal projekt, či bol projekt odovzdaný v danom termíne, aká verzia projektu bola odovzdaná, a iné

Užívateľ s rolou **Teacher** využíva webové stránky *Správa zápočtů* a *Projekty od studentů*. Funkcionalita je rovnaká ako pre užívateľskú rolu Admin.

Užívateľ s rolou **Student** využíva nasledovné webové stránky:

- *Přihlášení na zápočet* - študent sa môže prihlasovať alebo odhlasovať na termíny zápočtov daného predmetu
- *Upload projektů* - študent môže uploadovať projekt daného predmetu, pri uploade projektu tu prebieha aj kontrola, či užívateľ uploaduje projekt v správnom tvare, v povolenom termíne a či projekt obsahuje všetky potrebné súbory. Študent uploaduje projekt priamo svojmu vyučujúcemu do jeho zložky na serveri

5.1.2 Správa webovej stránky



Obr. 14: Správa webovej stránky

Na obrázku 14 je ukázané rozhranie na správu a úpravu webových stránok jednotlivých predmetov. Do textboxu administrátor môže písať aktualizácie pre daný predmet a po kliknutí na tlačítko *ODESLAT* sa novinka zobrazí na webovej stránke predmetu, ale súčasne sa novinka pošle aj študentom a vyučujúcim, prihláseným na odber noviniek pre daný predmet. Po každej aktualizácii stránky sa aktualizuje aj dátum a čas úpravy. Tlačítko *Upravit stránku* dovoľuje administrátorovi prejsť do HTML editoru a priamo upravovať aktuálnu stránku. Tlačítko *Správa souborů* presunie administrátora do rozhrania správcu súborov a automaticky mu zobrazí súborový priečinok upravovaného predmetu. Toto rozhranie je zobrazené na každej stránke predmetu. Ostatné webové stránky,

ako úvodná stránka portálu DB Education a sekcie *Studijní literatura*, *Oracle Academy* a *Software* obsahujú tiež toto rozhranie, ale bez textboxu na zasielanie noviniek.

5.2 Použité komponenty

Okrem už spomínaných komponent tretích strán(CKFinder a ELFinder), boli použité aj interné faktulné komponenty, ktoré pripravil Ing. Pavel Bednář. Pán Bednář vytvoril dve hlavné komponenty, veľmi dôležité pre funkcionalitu stránky: VsbLogin (LDAP prihlasovanie) a MailSender (komponenta na posielanie emailov).

5.2.1 VsbLogin

Komponenta VsbLogin je rozšírením rozhraní *Identity* a *IPrincial*, ktoré sú využívané na správu užívateľov ASP.NET frameworku. Táto komponenta umožňuje jednoduchý prístup do LDAP systému VŠB-TUO. Umožňuje prihlasovať užívateľov podľa interných univerzitných prihasovacích údajov a získať o nich informácie (meno, priezvisko, login, atď').

Komponenta poskytuje triedu *SecuredPage*, ktorá dedí z *asp WebPage* a poskytuje možnosť obmedzenia prístupu na stránku pre jednotlivé roly. Obmedzenie prístupu sa určuje podľa triedneho atribútu *RestrictedTo*

```
[RestrictedTo(Constants.RoleAdmin, Constants.RoleTeacher)]
public partial class ProjectManagement : SecuredPage
```

Výpis 1: Obmedzenie prístupu na webovú stránku

```
public string GetFullName(string login)
{
    VsbLogin.UserInfo userInfo = new VsbLogin.UserInfo(login);
    if (userInfo == null || !userInfo.IsLdapUser) return "";
    else return userInfo.FullName;
}
```

Výpis 2: Získanie celého mena pomocou komponenty VsbLogin

5.2.2 MailSender

Komponenta MailSender vytvárá jednotné rozhranie na posielanie emailov pomocou školského SMTP serveru. Umožňuje vytvárať hlavičku, telo emailu a odosielať emaily.

```
MailSender.MailSender mailSender = new MailSender.MailSender(login + "@vsb.cz");
string separator = "<ol_id=\"NewsList\">";

try
{
    MailSender.MailMessage email = mailSender.CreateMessage(mailSubject + " _ _ novinky",
        dbTemplate.GetHtmlBody(), recipients);
    email.IsBodyHtml = true;
    mailSender.SendMail(email);

    MailSender.MailMessage confirmation = mailSender.CreateMessage(mailSubject + " _ _ novinky",
        dbTemplate.GetHtmlBody(), (login + "@vsb.cz"));
    confirmation.IsBodyHtml = true;
    mailSender.SendMail(confirmation);
    AddNewsToContent(separator, newsMessage, false);
}
```

Výpis 3: Potvrdzovací email

Obrázok 15 ukazuje výsledný informačný email:



© 2015 DB Education

Obr. 15: Ukážka informačného emailu

5.3 Úprava zdrojového kódu ELFinderu

Jednou z požiadaviek na funkcionálnosť správcu súborov bola možnosť otvárať ho v konkrétnych podzložkách koreňového priečinku. Nanešťastie, používaná verzia ASP.NET ELFinder komponenty toto neumožňovala. Z toho dôvodu bola nutná úprava zdrojového kódu. Najväčším problémom bola skutočnosť, že JavaScript ELFinderu bol poskytnutý v minizovanej verzii - kód bol pre človeka nečitateľný. Bolo preto nutné stiahnuť si konkrétnu verziu plného JS z PHP implementácie ELFinderu. Následne bolo možné pridať do JS podporu otvárania v podzložkách. Spracovanie upravených AJAX requestov si vyžiadalo aj úpravu ELFinder konektoru (serverová časť komponenty). Ukážky upraveného kódu sú nižšie:

```
public Model.DirectoryModel GetRootDirectory(string openInSubfolder)
{
    string absolutePath = _config.LocalFSRootDirectoryPath;

    if (!string.IsNullOrEmpty(openInSubfolder))
    {
        absolutePath = Path.Combine(absolutePath, openInSubfolder);
    }

    DirectoryInfo di = getValidDirectoryInfo( absolutePath );
    if ( di == null )
        return null;

    return createDirectoryModel( di, EncodePathToHash( di.FullName ), null );
}
```

Výpis 4: Upravená metóda GetRootDirectory

Výpis 4 ukazuje zmenu otvárania root priečinku v prípade, že bol v javascripte vyplnený podpriečinok. Vo výpise 5 je ukázaný serverový handler, ktorý spracuje javascript request. Ak request obsahuje informáciu o podzložke, tak otvorí podpriečinok, inak defaultne otvorí koreňovú zložku.

```

public Response.IResponse Execute(CommandArgs args)
{
    ...

    if (oa. init .IsTrue())
    {
        return new Response.InitDirectoryResponse(_config.ApiVersion,
        new string[0],
        _config.UploadMaxSize,
        cwd,
        subItems.ToArray(),
        new Model.OptionsModel(vol.GetPathToRoot(cwd), _config.BaseUrl, _config.BaseThumbUrl)
        );
    }
    else
    {
        return new Response.DirectoryResponse(cwd,
        subItems.ToArray(),
        new Model.OptionsModel(vol.GetPathToRoot(cwd), _config.BaseUrl, _config.BaseThumbUrl)
        );
    }
}

```

Výpis 5: Upravená metóda v triede OpenCommand

5.4 Udalosti v Menu.ascx

Celý informačný systém je úzko prepojený s databázou SQLite. Aby sa predišlo viacnásobným pripojeniam na databázu z rôznych web stránok, jednotlivé podstránky majú prístup k spoločnej vlastnosti (property) z používanej masterpage.

```

string connStr = ConfigurationManager.AppSettings["ConnectionString"];
private SQLiteConnection _conn;

public SQLiteConnection conn
{
    get
    {
        if (_conn == null)
        {
            _conn = new SQLiteConnection(connStr);
        }
    }
}

```

```
        _conn.Open();
    }

    return _conn;
}
}
```

Výpis 6: Spojenie s databázou

Problém bol s komponentami ASP Web Control (.ascx). Tieto komponenty nemajú prístup k masterpage. Na webovom portáli DB Education ide konkrétne o komponentu hlavného menu. Zoznam vyučovaných predmetov sa nahráva z SQLite databázy. Aby sme predišli duplicitnému spojeniu, boli do menu komponenty pridané vlastné udalosti (eventy). Všetky fungujú na podobnom princípe, teda stačí, ak opíšeme jednu z nich.

```
public event Action OnDisplayCourses;

protected void Page_Load(object sender, EventArgs e)
{
    ...

    if (OnDisplayCourses != null)
    {
        OnDisplayCourses();
    }

    ...
}
```

Výpis 7: Volanie udalosti v Menu.ascx

Výpisy 7 a 8 spolu veľmi úzko súvisia. Pri vykresľovaní sekcie s vyučovanými predmetmi sa zavolá udalosť OnDisplayCourses. Na túto udalosť je v masterpage naviazaná metóda (Výpis 8), ktorá vyselektuje predmety pre daný ročník. Kompletnú sekciu s požadovanými dátami pošle späť komponente Menu.ascx.

```
protected void Menu1_OnDisplayCourses()
{
    if (Request["year"] != null && !Request.Url.ToString().ToLower().Contains("
        subjectmanagement"))
    {
        string oldYear = Request["year"].ToString();
        List<Subject> allSubjects = Subject_Gateway.Select(conn, oldYear);
        List<Subject> ceske = allSubjects.Where(sub => sub.Cz_En == "cz").ToList();
        Menu1.DataBindCourses(ceske, "cz");

        List<Subject> anglicke = allSubjects.Where(sub => sub.Cz_En == "en").ToList();
        Menu1.DataBindCourses(anglicke, "en");
    }
    else
    {
        List<Subject> allSubjects = Subject_Gateway.Select_Actual(conn);
        List<Subject> ceske = allSubjects.Where(sub => sub.Cz_En == "cz").ToList();
        Menu1.DataBindCourses(ceske, "cz");

        List<Subject> anglicke = allSubjects.Where(sub => sub.Cz_En == "en").ToList();
        Menu1.DataBindCourses(anglicke, "en");
    }
}
```

Výpis 8: Naviazaná metoda z masterpage

5.5 Kontrola odosielaných súborov v .zip archíve

Jednou z hlavných funkcií, je odovzdanie projektu do IS. Projekty sa odosielajú ako archív .zip. Systém je schopný automaticky porovnať odoslané súbory s požiadavkami projektu pre dané zadanie. Vyučujúci majú tri možnosti ako vytvoriť zadanie projektu:

1. Nekontrolovať obsah .zip archívu
2. Kontrolovať iba názvy súborov
3. Kontrolovať názvy aj formát súborov

Metóda `UnzipAndCheck` vytvorí v databáze záznam o nahranom archíve a následne vytvorí zoznamy súborov, ktoré sú navyše, a ktoré v archíve chýbajú. Veľkým pomocníkom pri práci so zoznamami, je v C# Linq, Lambda funkcie a rozširovacie metódy. Všetky spomenuté prvky sú použité aj v metóde `AdjustSuffixExcept`, ktorá porovnáva dva zoznamy a prihliada na používanie/nepoužívanie formátov súborov.

```
private void UnzipAndCheck(string savePath, bool overTime)
{
    ...

    using (Ionic.Zip.ZipFile zip = Ionic.Zip.ZipFile.Read(savePath))
    {
        List<string> required = new List<string>();
        List<string> provided = new List<string>();
        List<string> missing = new List<string>();
        List<string> excessive = new List<string>();

        required = proj.Specifikacia.Split(';').Where(subor => !String.IsNullOrEmpty(subor.
            Trim())).Select(subor => subor.ToLower()).ToList();

        provided = zip.EntryFileNames.Select(subor => subor.ToLower()).ToList();

        missing = required.AdjustSuffixExcept(provided, false);
        excessive = provided.AdjustSuffixExcept(required, true);

        ...
    }
}
```

Výpis 9: Časť metódy `UnzipAndCheck`

Kód vo Výpise 9 rozparsuje nahraté súbory a špecifikáciu projektu, vytvorí 4 zoznamy (požadované, poskytnuté, chýbajúce a prebytočné) a zavolá metódu `AdjustSuffixExcept` vo Výpise 10. Metóda `AdjustSuffixExcept` porovná nahraté a požadované súbory a podľa príznaku buď prihliada, alebo neprihliada na formát súboru. Potom už iba metóda `AdjustSuffixExcept` naplní zoznamy prebytočných a chýbajúcich súborov vo Výpise 9.

```

public static List<string> AdjustSuffixExcept(this List<string> Input, List<string> Second, bool
    AdjustSuffixBySecondList)
{
    if (Input == null || Second == null) return null;

    List<string> Result = new List<string>();

    if (AdjustSuffixBySecondList)
    {
        Result = Input;

        foreach(string Item in Second)
        {
            if (Item.Contains("."))
            {
                Result.RemoveAll(text => text == Item);
            }
            else
            {
                Result.RemoveAll(text => Path.GetFileNameWithoutExtension(text) == Item);
            }
        }
    }
    else
    {
        foreach (string Item in Input)
        {
            if (Item.Contains("."))
            {
                if (Second.All(text => text != Item))
                {
                    Result.Add(Item);
                }
            }
            else
            {
                if (Second.GetWithoutExtension().All(text => text != Item))

```

```
        {  
            Result.Add(Item);  
        }  
    }  
}  
  
return Result;  
}
```

Výpis 10: Porovnanie požadovaných súborov

6 Záver

Táto práca povýšila jednoduchý webový portál DB Education na moderný informačný systém. Nutnou prerekvizitou správneho vypracovania zadanej témy bolo naštudovanie funkcionality pôvodnej webovej stránky. Do nej boli implementované už existujúce komponenty (ako z dielne fakulty, tak aj z tretích strán). Stránka bola taktiež rozšírená o databázu SQLite a množstvo nových funkcií. V informačnom systéme sú implementované štyri roly (administrátor, učiteľ, študent, anonymný hosť). Z toho však rola učiteľa je momentálne nevyužívaná, ale plne funkčná a pripravená k použitiu v budúcnosti.

Z vlastného pohľadu bakalársku prácu hodnotím ako úspešne splnenú. Všetky požiadavky na vypracovanie práce boli dokončené. Prerobený webový portál spolu s novými komponentami tvoria komplexný funkčný celok.

Samozrejme portál je možné ešte ďalej rozširovať. Jedným z možných rozšírení je napríklad implementácia algoritmu, ktorý by automaticky vyhodnocoval študentmi odovzdané projekty. Dalším možným vylepšením by mohla byť implementácia komponenty, ktorá by bola prepojená s databázou a študenti by písali testy priamo cez portál DB Education. Rovnako treba podotknúť, že stránka nie je prispôsobená pre mobily a tablety. Existujú dve riešenia: buď vytvoriť plne responzívny web, alebo vytvoriť mobilnú aplikáciu.

7 Literatúra

- [1] CKEditor [online]. [cit. 2015-04-13]. Dostupné z: <http://ckeditor.com/>
- [2] CKEditor [online]. [cit. 2015-04-25]. Dostupné z: <http://en.wikipedia.org/wiki/CKEditor>
- [3] CKSOURCE. *CKFinder* [online]. [cit. 2015-04-13]. Dostupné z: <http://cksource.com/ckfinder/features>
- [4] CodePlex [online]. [cit. 2015-04-25]. Dostupné z: <https://www.codeplex.com/>
- [5] CODEPLEX. *XML Editor - A Framework and Developer Toolkit* [online]. [cit. 2015-04-13]. Dostupné z: <http://xmlwebpad.codeplex.com/>
- [6] CuteEditor [online]. [cit. 2015-04-13]. Dostupné z: <http://cutesoft.net/ASP.NET+WYSIWYG+Editor/>
- [7] DEVEXPRESS. *ASP.NET AJAX Control Toolkit* [online]. [cit. 2015-04-13]. Dostupné z: <http://ajaxcontroltoolkit.codeplex.com/>
- [8] ELEMENT IT. *HTTP Commander* [online]. [cit. 2015-04-13]. Dostupné z: <http://www.element-it.com/asp-net-explorer-browser/online-share/web-file-manager.aspx>
- [9] ELRTE [online]. [cit. 2015-05-04]. Dostupné z: <http://elrte.ru/>
- [10] GITHUB. *ElFinder* [online]. [cit. 2015-04-13]. Dostupné z: <https://github.com/Studio-42/ELFinder>
- [11] KREIBICH, Jay A. 2010. *Using SQLite*. 1st ed. Beijing: O'Reilly, xx, 503 s. ISBN 978-059-6521-189.
- [12] MACDONALD, Matthew. 2012. *Beginning ASP.NET 4.5 in C#*. New York: Distributed to the book trade worldwide by Springer Science Business Media New York, xxxvi, 885 p. Expert's voice in .NET. ISBN 14-302-4251-5.
- [13] Rich Text Editor [online]. [cit. 2015-04-13]. Dostupné z: <http://richtexteditor.com/>
- [14] SharePoint [online]. [cit. 2015-05-04]. Dostupné z: <http://www.zive.cz/clanky/neni-sharepoint-jako-sharepoint/sc-3-a-163574/>
- [15] SQLite: Databáze pro váš web [online]. [cit. 2015-04-21]. Dostupné z: <http://www.zdrojak.cz/clanky/sqlite-database-pro-vas-web/>

- [16] TELERIK. *RadEditor* [online]. [cit. 2015-04-13]. Dostupné z: <http://www.telerik.com/products/aspnet-ajax/editor.aspx>
- [17] TINY MOXIECODE CONTENT EDITOR. *TinyMCE* [online]. [cit. 2015-05-04]. Dostupné z: <http://www.tinymce.com/>
- [18] THE OPEN WEB APPLICATION SECURITY PROJECT. *Testing for AJAX Vulnerabilities* [online]. [cit. 2015-04-13]. Dostupné z: https://www.owasp.org/index.php/Testing_for_AJAX_Vulnerabilities
- [19] Úvod do ASP.NET [online]. [cit. 2015-04-21]. Dostupné z: <http://www.itnetwork.cz/tutorial-uvod-do-asp-dot-net>
- [20] WebDAV [online]. [cit. 2015-05-04]. Dostupné z: <http://en.wikipedia.org/wiki/WebDAV>

A Obsah priloženého CD

1. Zdrojový kód DB Education
2. Tento dokument vo formáte .PDF